info@inc4.net
inc4.net
INC4

Emerald Circuit contracts Core Audit

Dated: 31 March, 2022
Authored: by Andrii Rozinko

1. Introduction

The contracts are hosted at:

https://github.com/emerald-x/contract

All the contracts in the "contracts" folder are in scope.

The git commit hash evaluated: 6a9896589c494929646febeaf8945380bcb37b39

2. Disclaimer

The audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, regulatory regime for the business model, or any other statements about fitness of the contracts to purpose, or their bug-free status. The audit documentation is for discussion purposes only.

3. Discussion

- Standalone uses of the OpenZeppelin sources

This project uses the OpenZeppelin sources in the wrong way – by including a copy of the source files in the project's source tree. This use of OpenZeppelin makes it difficult to update these sources. We recommend using this as standard – as an independent library.

4. Affected files:

contracts/dependencies:

Context.sol
ERC165.sol
ERC721Enumerable.sol
ERC721.sol
Ownable.sol

contracts/dependencies/utils:

Address.sol
Initializable.sol
Strings.sol

contracts/interfaces:

IERC165.sol
IERC20.sol
IERC721Metadata.sol
IERC721Receiver.sol
IERC20Metadata.sol
IERC721Enumerable.sol
IERC721Project.sol
IERC721.sol

- Insignificant description of interfaces

The interfaces declared in the project are not well documented. It is desirable to add a little description of the purposes of defining these interfaces, as well as a description of the events and methods of these interfaces.

Affected files:

contracts/interfaces:

IDomainRegistrar.sol
IMarketplace.sol
IProjectsFactory.sol
IRegistry.sol

- Run tests issue

The non-standard name of the folder with tests ("tests" instead of "test") does not allow running tests with the script command "test" (yarn test).

5. Line by Line Comments

- contracts/domains/DomainRegistrar.sol

line 26: line 31: line 78: line 79:
require(registry.owner(baseNode) == address(this));
require(_controller == msg.sender);
require(available(id));
require(block.timestamp + duration + GRACE_PERIOD > block.timestamp + GRACE_PERIOD); // Prevent future overflow

line 98: require(expiries[node] + GRACE_PERIOD >= block.timestamp); // Name must be registered here or in grace period

line 99: require(expiries[node] + duration + GRACE_PERIOD > duration + GRACE_PERIOD); // Prevent future overflow

line 110: require(_isApprovedOrOwner(msg.sender, id));

All "require" function must have an error string.

- contracts/domains/DomainsController.sol

line 80: require(msg.value >= cost);

All "require" function must have an error string.

- contracts/domains/DomainsController.sol

line 24: require(_controller == msg.sender);

All "require" function must have an error string.

- contracts/domains/Registry.sol

line 20: require(owner == msg.sender);

All "require" function must have an error string.

- contracts/space/Marketplace.sol

line 28: mapping(address => mapping(uint256 => ListingInfo)) internal listingInfos;

The variable name "listingInfos" does not follow the naming convention for "internal" variables - it should be "_listingInfos".

line 366: emit ListingStarted(collection, tokenId, listing.node, listing.listingType, listing.minimalBid, block.number);

The ListingStarted event is generated not only to start the listing, but also to stop the listing. To determine the reason for emitting events, you must use an additional call to the getListingStatus method of the Marketplace contract. I recommend introducing the ListingStop event or using custom event parameter values to specify the reason ("stopListing") for the event to occur.

line 375: emit UnreturnedBidsClaimed(msg.sender, amount); This event fires regardless of the send status.

6. Implemented changes validation

Dated: 20 April, 2022
Authored: by Andrii Rozinko

All recommended changes have been implemented accordingly.

Validation result: positive.